Cedutech中教仪®

智能灭火机器人说明书

目录

1. 智能灭火机器人简介

智能灭火机器人是一个综合性的机器人平台，使用了多种数字量、模拟量、I2C 的传感器和高集成度的 MJ3100 控制器，体现了机器人的复杂性和趣味性，充分展示了平台的多功能性和扩展性。

本篇制作的智能灭火机器人功能有：

1. 安全行驶功能；能在白色桌面上循边行驶而不会掉落。
2. 智能灭火功能；一旦发现桌面中央有火焰情况，能迅速移动靠近火焰，及时吹灭火焰。

使用到的传感器和电子件有：

1. 无线遥控一套
   作用：控制机器人的起停。
2. LED 显示屏一块
   作用：显示机器人的状态。
3. 灰度传感器两个
   作用：判断白色桌面的边缘。
4. 红外测距传感器一个
   作用：测量机器人距离火源的距离
5. 火焰传感器两个
   作用：监控发现火源
6. LED 灯一个
   作用：火源报警指示灯
7. 三轮小车平台一个

2. 智能灭火机器人搭建

下面开始组装智能灭火机器人：



图 1 三轮小车平台

如图 1，首先搭建好三轮小车平台，使用了一个 18X18 的孔板和一对电机和一个万向轮。在小车的右前方安装一个火焰传感器，用来探测火源。如图 2 所示：

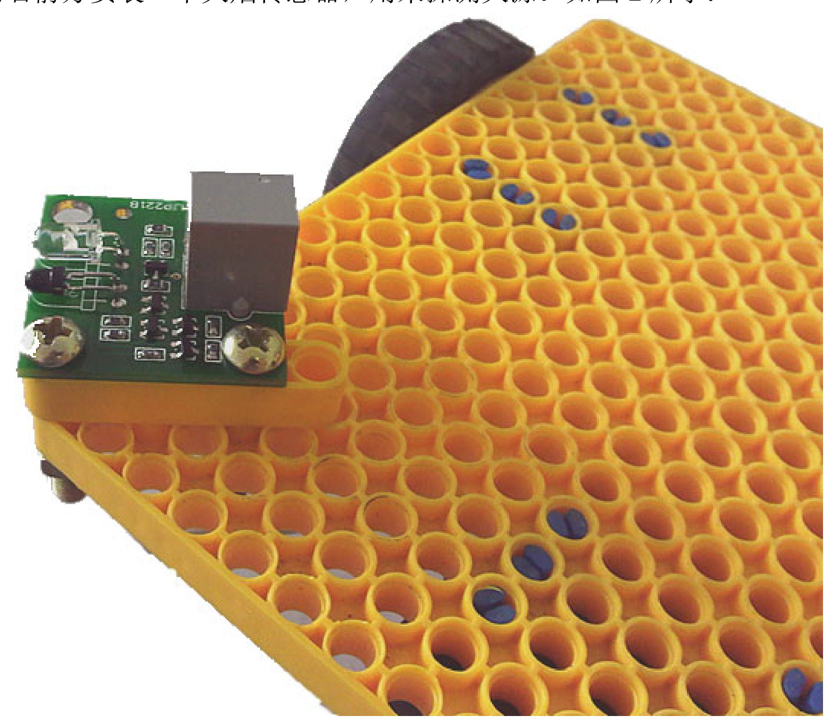


图 2 安装右前方的火焰传感器

接着在小车的前方安装一个红外测距传感器，用来灭火的时候测量机器人距离火源的距离。如图 3，先给红外测距传感器安装孔杆。

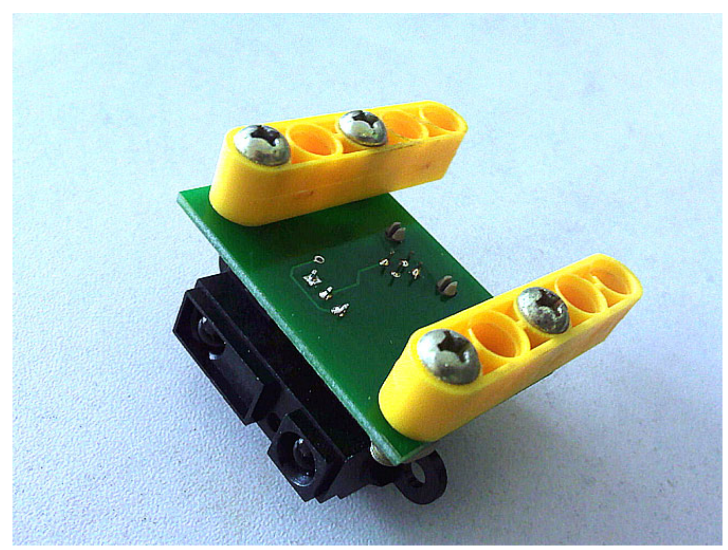


图 3 给红外测距传感器安装孔杆

用一个两孔有肩插销将之安装到小车前面，如图 4

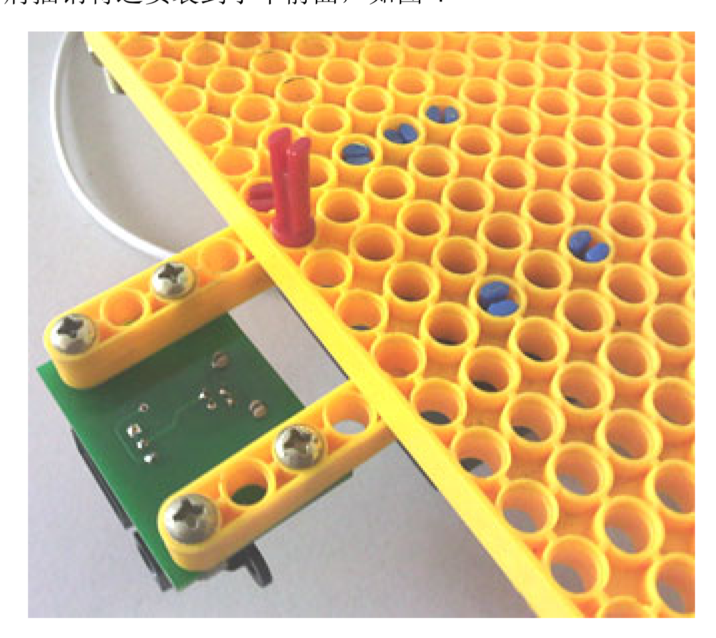


图 4 小车前方安装红外测距传感器

当小车灭火的时候，前面需要有火焰传感器监控灭火情况，故在小车前面安装另一个火焰传感器，如图 5

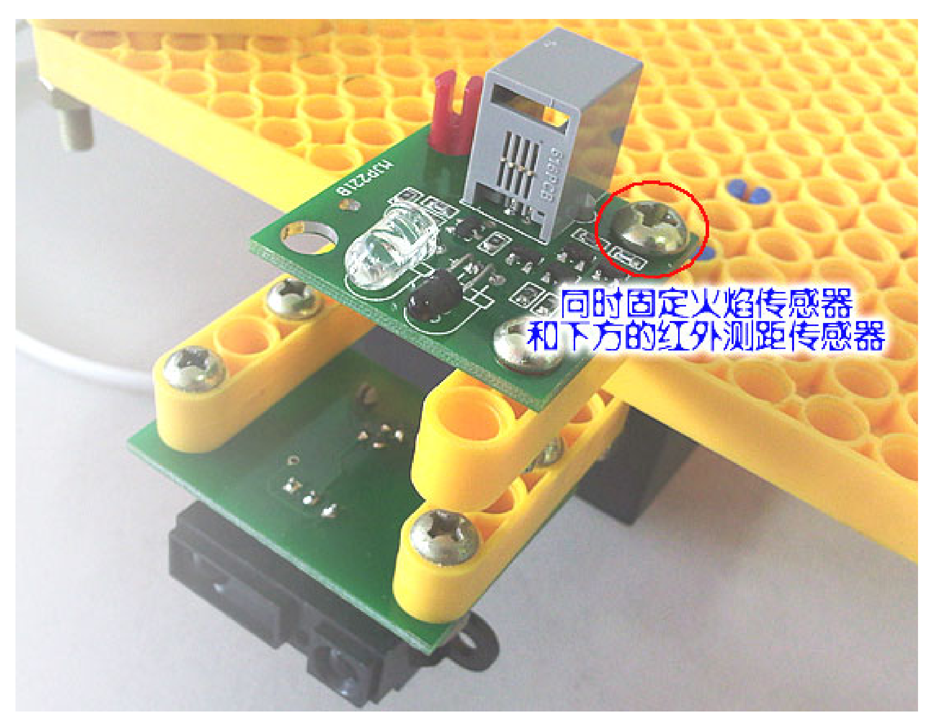


图 5 安装前面的火焰传感器

在实际应用中，灭火机器人需要安装灭火专用电机和灭火风扇。本篇中只是示例性的安装。首先搭建风扇支架；如图 6：

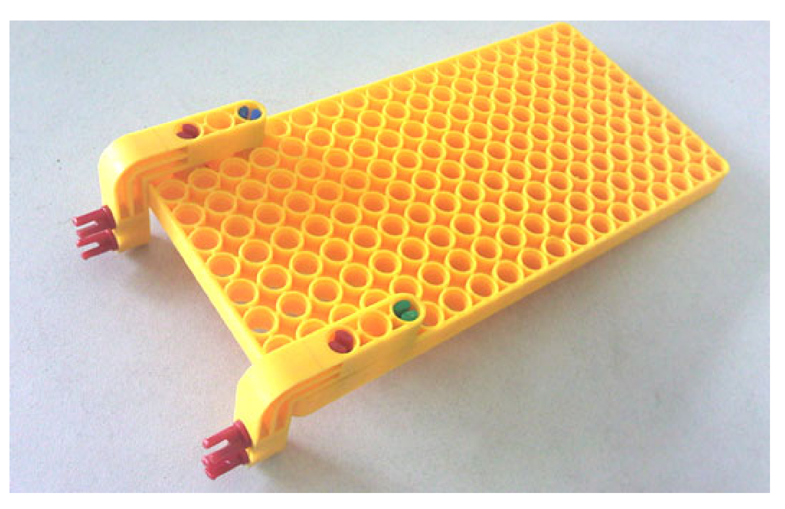


图 6 给灭火风扇搭建支架

接着将支架安装到小车平台上，如图 7：

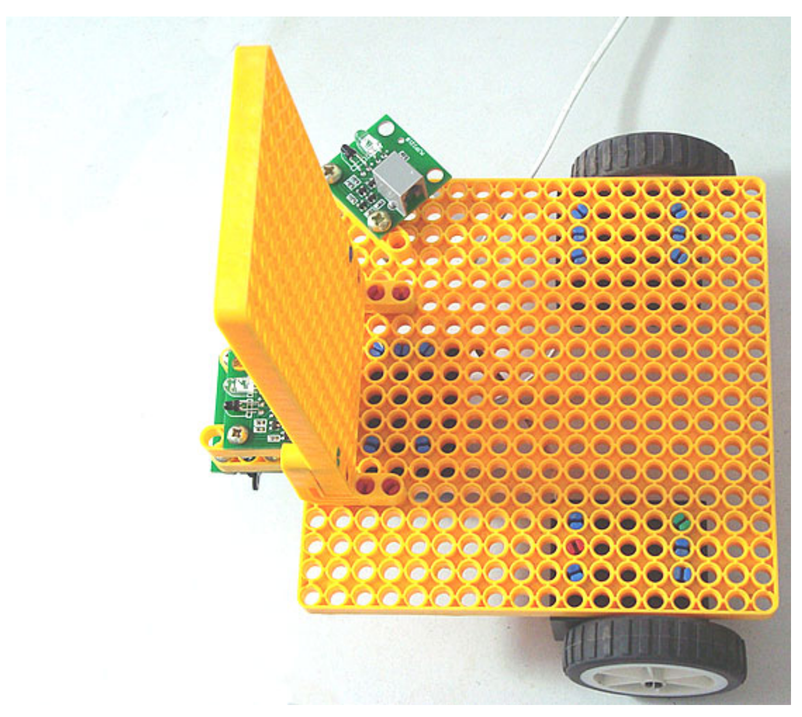


图 7 安装风扇支架

接着在支架上安装电机，如图 8：

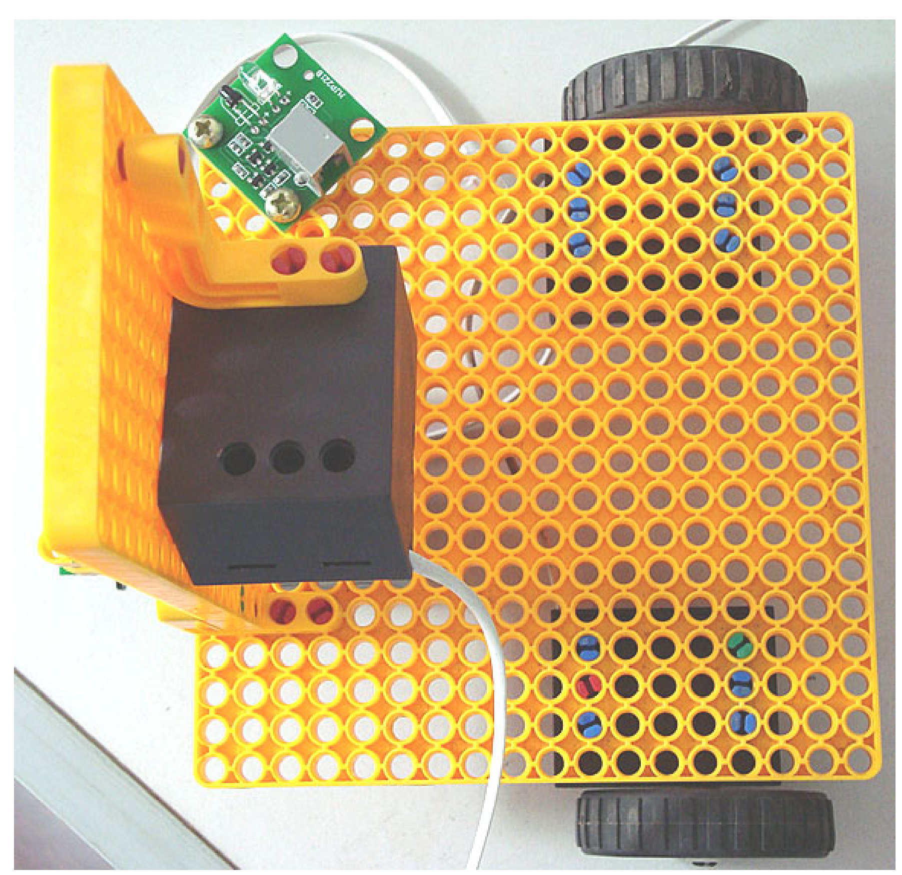


图 8 在支架上安装风扇电机

再安装风扇，如图 9：

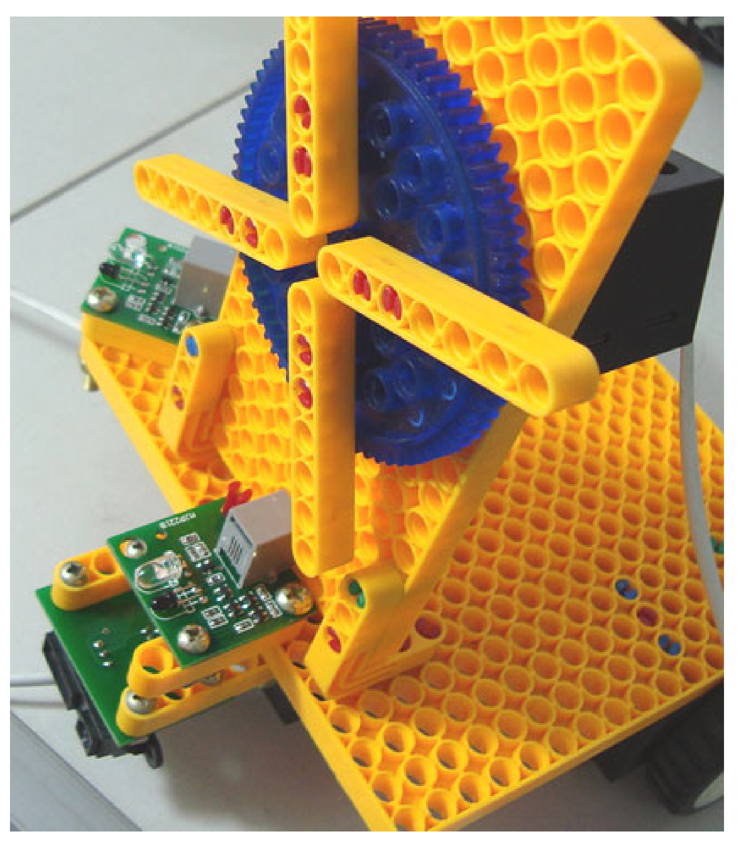


图 9 安装风扇

小车具有安全行驶的功能，这里的行驶是小车在桌面边缘沿着顺时针方向行驶。所以需要在其左侧安装灰度传感器。先在左前方安装一个灰度传感器，如图 10：

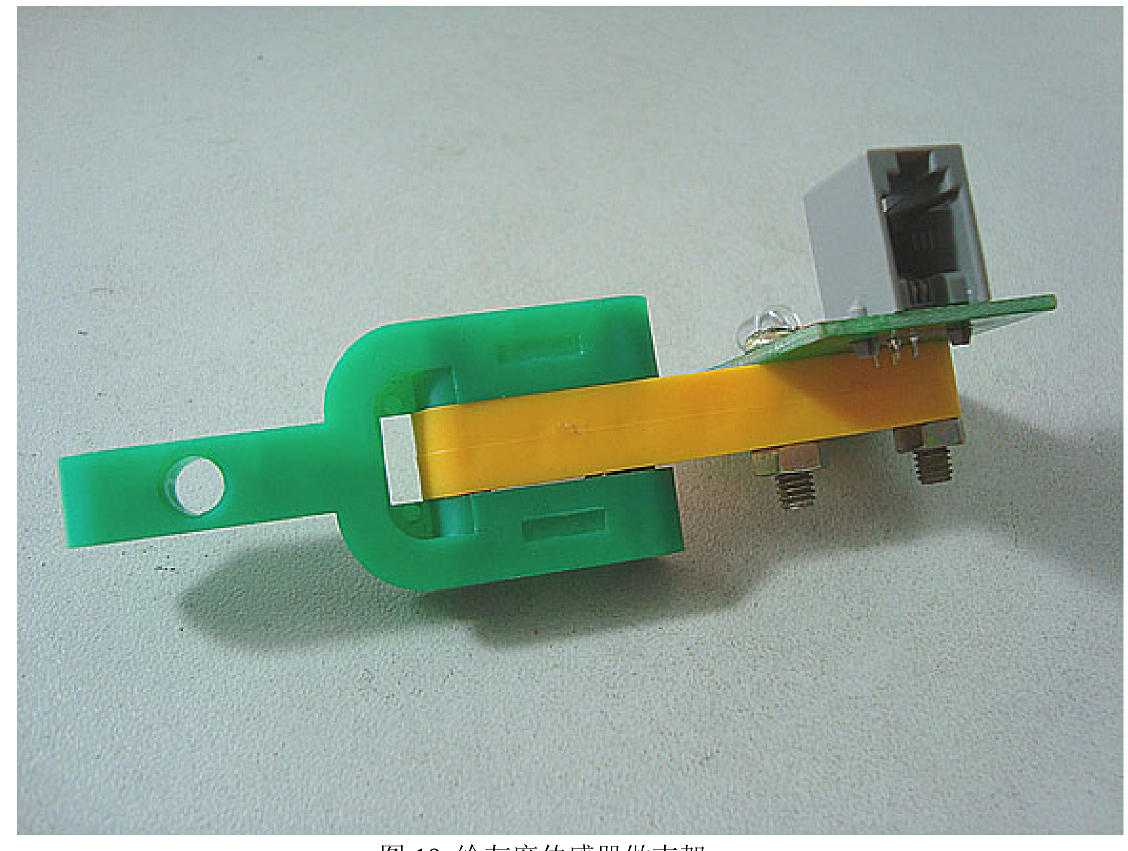


图 10 给灰度传感器做支架

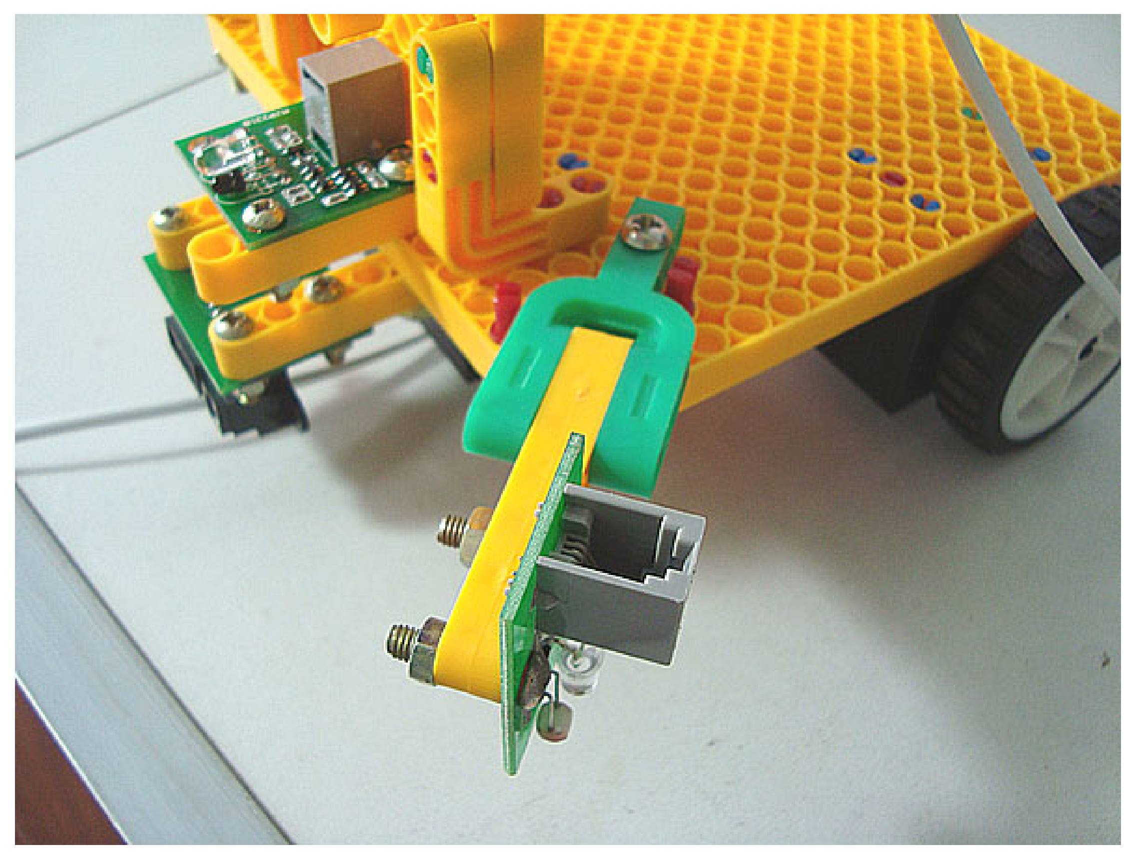


图 11 安装左前方灰度传感器

同样在小车的左侧安装一个灰度传感器，如图 12：

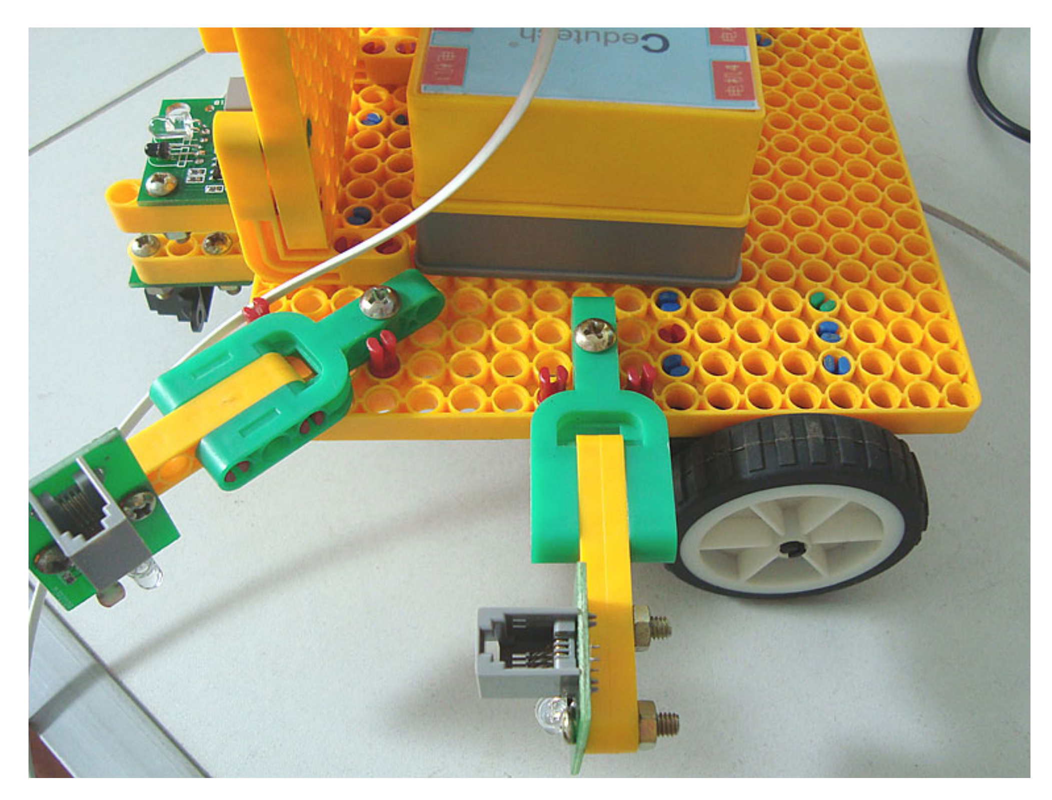


图 12 安装小车左侧的灰度传感器

接着给小车安装 LED 显示屏，如图 13

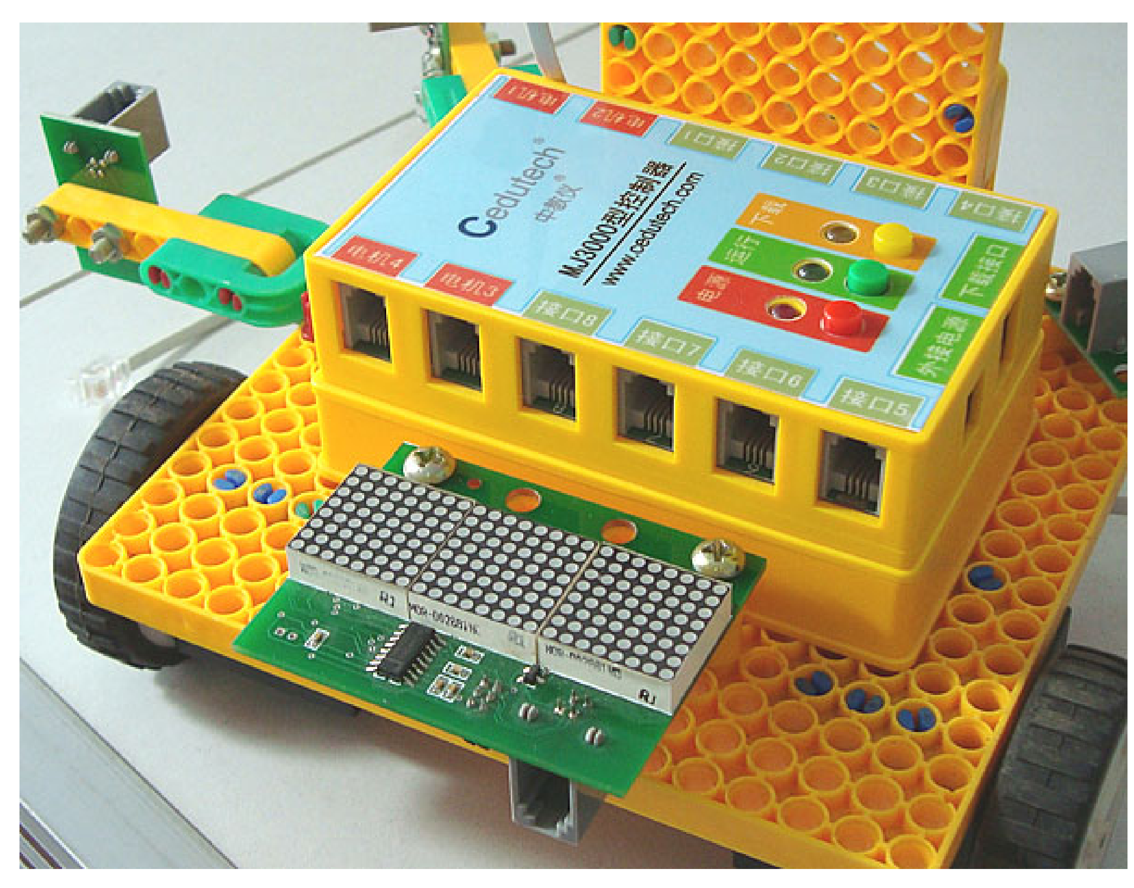


图 13 在小车尾部安装 LED 显示屏

接着给机器人安装无线遥控接收器，如图 14：

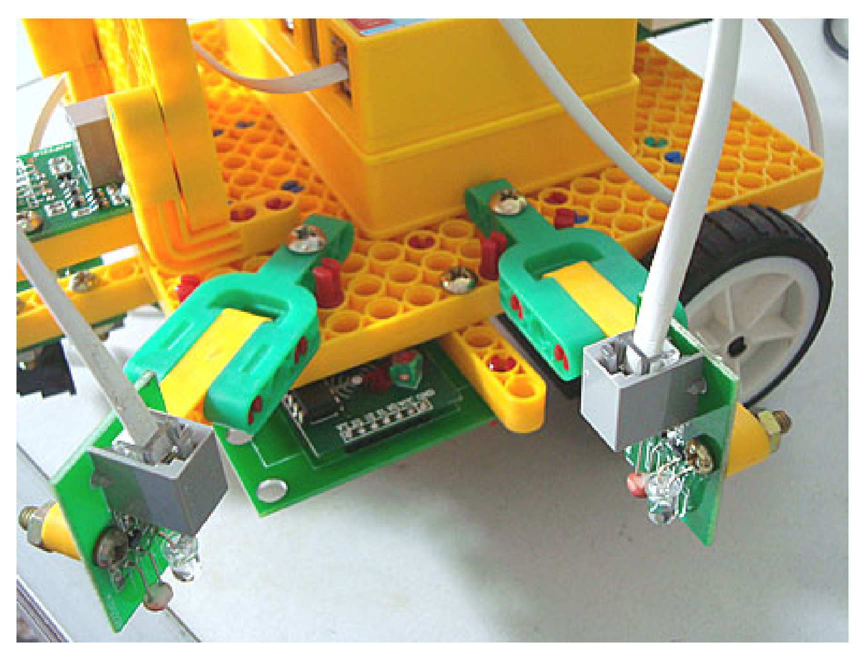


图 14 安装无线遥控接收器

最后给各个电子件接线；那么整个机器人全貌是：

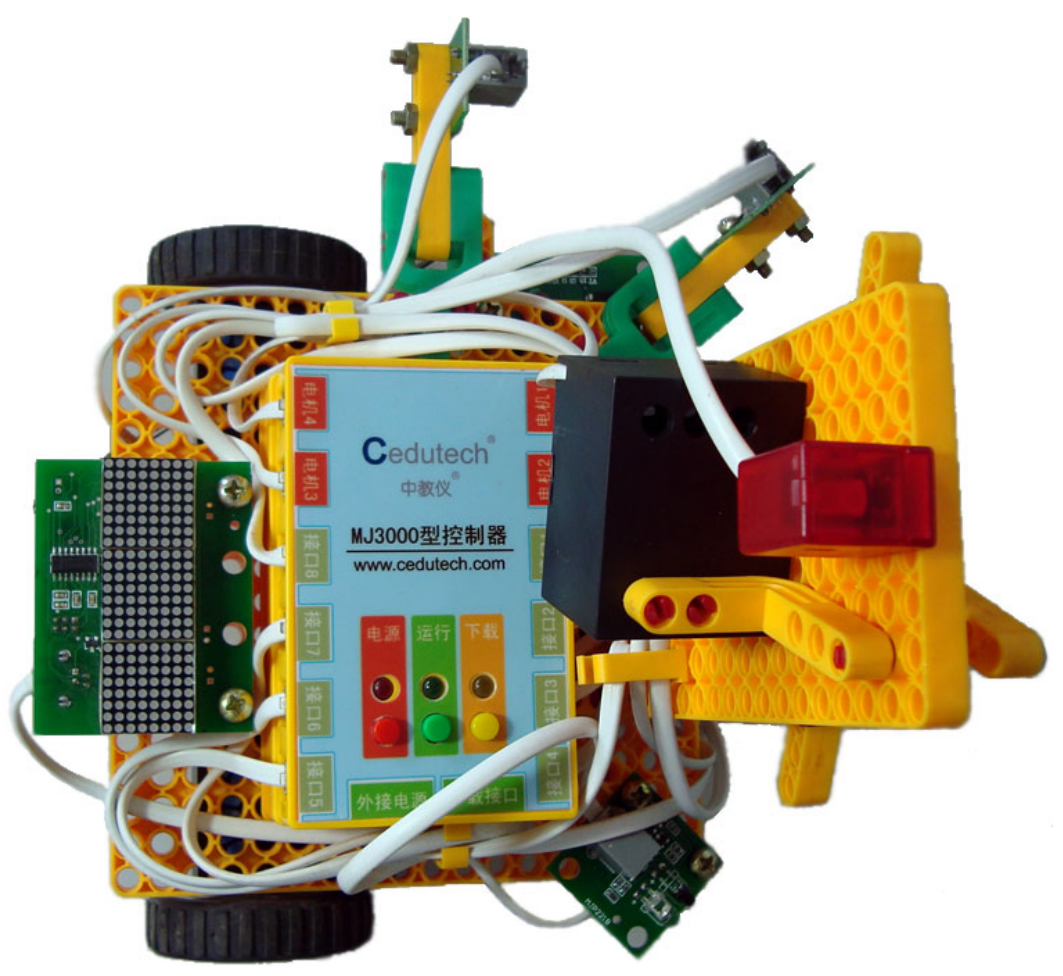


图 15 智能灭火机器人全貌

至此，机器人的装配完毕。我们总结一下机器人的功能和各个传感器、电子元件的用法。

机器人能沿着白色桌面边缘顺时针方向行走，故左前方和左侧的两个灰度传感器用来探测桌面边缘。机器人需要检测桌面中央的火焰，所以右前方的火焰传感器用来检测桌面中央的火焰。而前方的火焰传感器和红外测距传感器分别用来迎着火焰前进时检测火焰和测量距离使用。后面的 LED 显示屏和风扇支架上的红色 LED 灯用来反应机器人状态。

本篇中机器人的接线端口如图 16 所示：

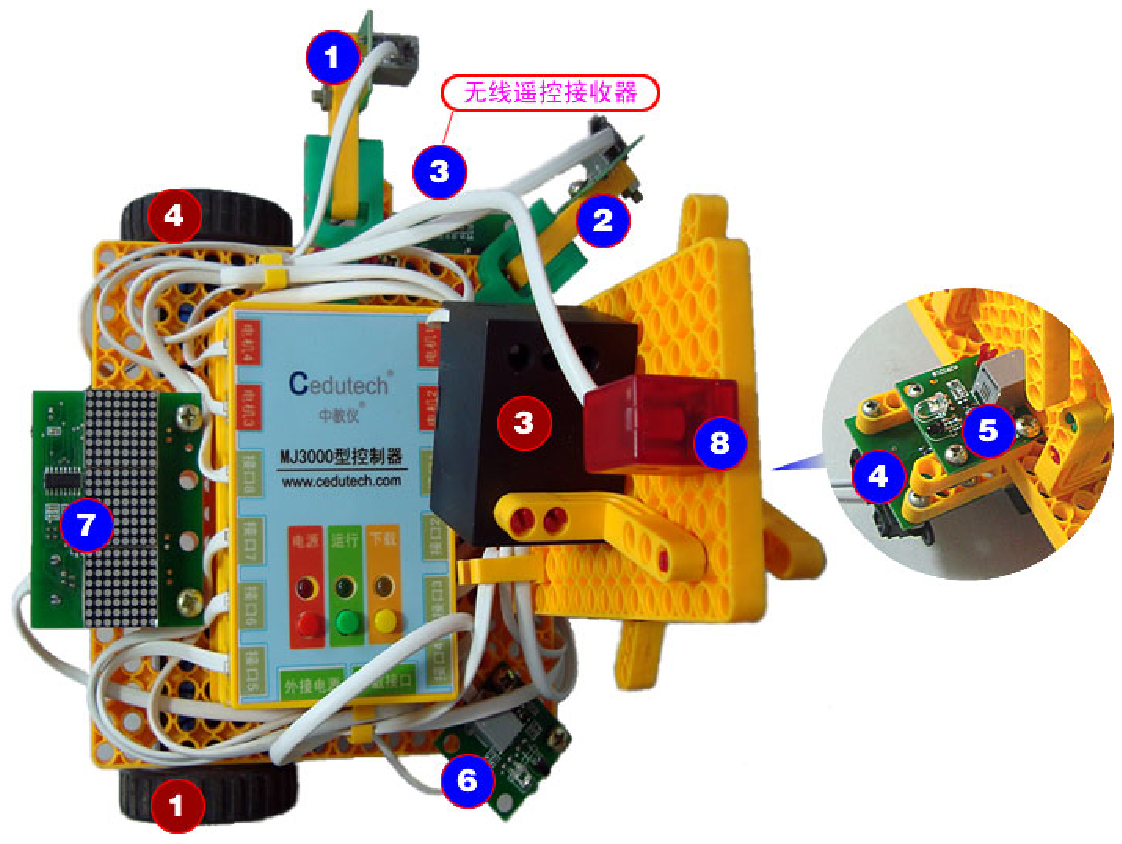


图 16 端口接线图

至此智能灭火机器人装配完毕。

3. 配套程序

接线端口如图所示的智能灭火机器人的配套程序如下：

```
/* ------端口设置自动生成代码------ */
#define HWPORT "aabaaabo"

/* ---------设备头文件------------- */
#include "MJ3100.h"

//  小车前进
void Cgo(unsigned char i){
	MotorRun(1, i);
	MotorBack(4, i);
}

//  小车后退
```

```
void Cback(unsigned char i){
    MotorRun(4, i);
    MotorBack(1, i);
}

// 小车停止
void Cstop(){
    MotorStop(1);
    MotorStop(4);
}

// 小车左转
void Cleft(unsigned char i){
    MotorRun(1, i);
    MotorRun(4, i);
}

// 小车右转
void Cright(unsigned char i){
    MotorBack(1, i);
    MotorBack(4, i);
}
void show(unsigned char a,unsigned char b,unsigned char c){
    SendLEDChar(7,1,a);
    SendLEDChar(7,2,b);
    SendLEDChar(7,3,c);
}
void alert(){
    LEDOn(8);
    show('S','O','S');
    I2C_ByteWrite(7 , 0x0A,0x04 ,1 );
}
void unalert(){
    LEDOff(8);
    show('G','o','.');
    I2C_ByteWrite(7 , 0x0A,0x04 ,0 );
}
void blow(){
    MotorRun(3,10);
}
void unblow(){
    MotorStop(3);
}
int main()
```

```
{
	/* ---------设备初始化-------------- */
	HWInit(HWPORT);
	/*
	Moto:
		1-右侧电机
		4-左侧电机
		3-风扇电机
	IO:
		1-左侧灰度
		2-左前灰度
		3-左下  无线遥控
		4-前方红外测距
		5-前方火焰
		6-右前火焰
		7-后方 Led 显示屏
		8-红色 Led 灯
	*/
	/* ---------用户代码开始  ----------- */
	LEDOff(8);show('H','i','!');
	while(!(GetRFData(3) & 8)){} //等待无线遥控开启运行

	PhotoSensorLEDOn(1);
	PhotoSensorLEDOn(2);
	PhotoSensorLEDOff(5);
	PhotoSensorLEDOff(6);
	unsigned char a4;
	unsigned int i,a1,a2,a5,a6,c1,c2,fire;
	while(1)
	{
		// 循环代码
		a1=GetAD(1);			WaitMS(5);
		a2=GetAD(2);			WaitMS(5);
		a4=GetIRDistance(4);		WaitMS(5);
		a6=GetAD(6);			WaitMS(5);
		a5=GetAD(5);

		//灭火，要求桌面中央蜡烛火焰至少 2CM 高度
		if(a6<320){//为右前火焰传感器发现火焰的阈值，需调试获得
			alert();
			while(a6>90){//右前传感器正对火焰的阈值，需调试获得
				a6=GetAD(6);
				Cright(7);
			}
```

```
        while(a5>40){//前方传感器正对火焰的阈值，需调试获得
            a5=GetAD(5);
            Cright(5);
        }
        while(GetIRDistance(4)>120 && GetAD(5)>35){//接近火焰的参数，需调试获得
            Cgo(7);
        }
        Cstop();
        blow();
        WaitMS(500);
        while(a5<50){//火焰存在，灭火
            a5=GetAD(5);
            if(i<3) Cright(5);
            else if(i>=3 && i<7) Cleft(5);
            else i=0;
            i++;
        }
        Cleft(5);
        WaitMS(500);
        unblow();
        unalert();

        show('O','K','!');
        while(1){
            Cstop();  //灭火结束
        }

    }
    else{//边缘行驶
        if(a1<330 || a2<330){//越出边缘时候左边和左前灰度传感器的阈值，需调试获得
            LEDOn(8);
            Cback(9);
            WaitMS(8);//刹车
            Cright(6);
            show('R','i','g');
        }
        else{
            show('G','o','.');
            Cgo(7);
            LEDOff(8);
        }
    }
```

```
        }

        return 0;
}
```

4. 调试传感器阈值

本机器人程序需要调试获得的参数有左侧和左前的灰度传感器；右前和前方的火焰传感器；前方红外测距传感器。

首先利用 RobotDC 下载测试内核；如图 17：

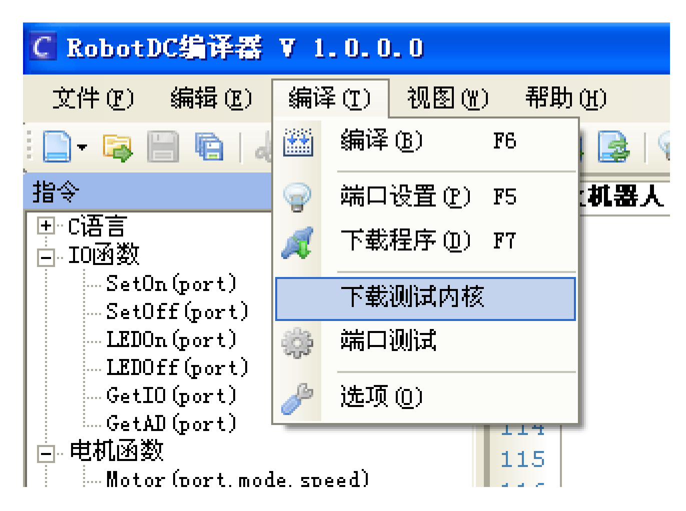


图 17 下载测试内核

下载完测试内核，即可进入端口测试；如图 18：

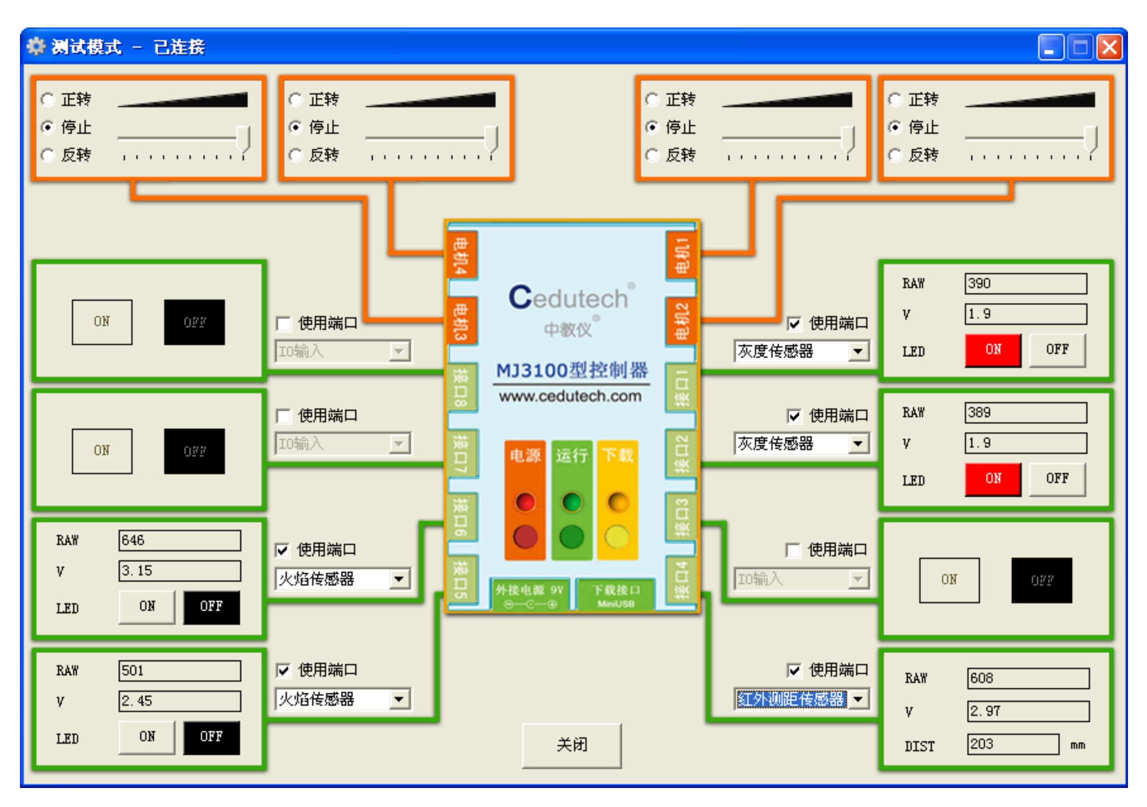


图 18 端口测试

分别将需要测试的传感器对应的端口设置好。将 1,2 端口的灰度传感器分别对着桌面和桌面外侧，记录灰度传感器越出桌面时的数值；即为 a1,a2 的阈值。如图 19；

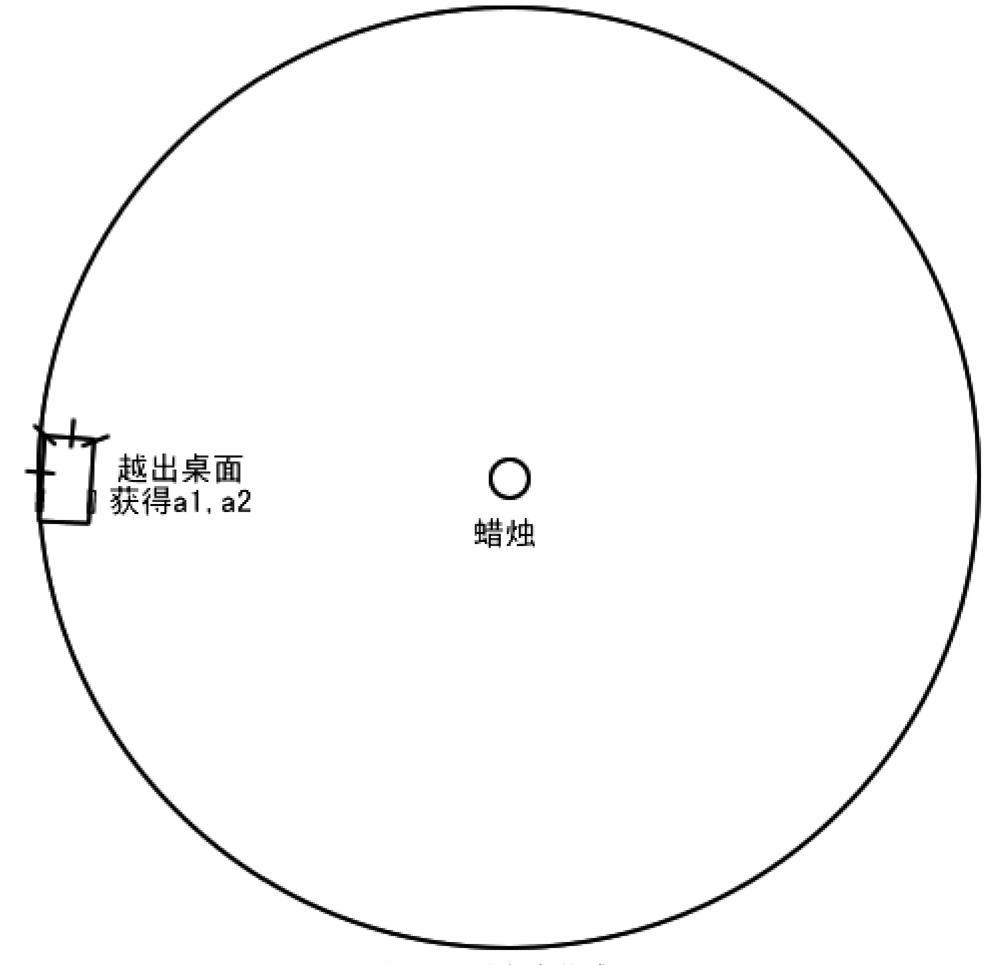


图 19 调试灰度传感器

将桌面中间的蜡烛点燃，并且保证蜡烛火焰高度在 2CM 左右，此时调试右前火焰传感器数值的变化，获得 a6 发现火焰的阈值；再将右前火焰传感器正对火焰，获得 a6 正对火焰时的阈值；同样以此获得 a5 正对火焰的阈值。如图 20；

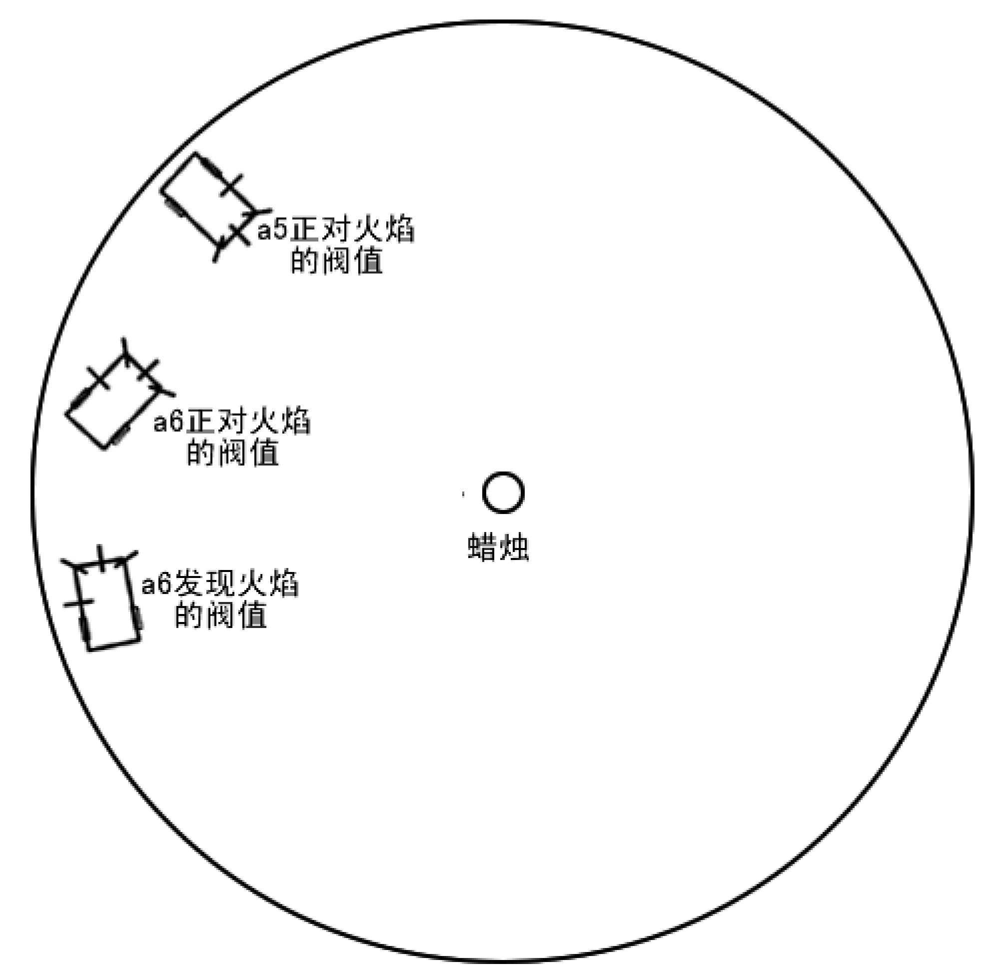


图 20 调试右前和前方的火焰传感器阈值

最后调试灭火时候的阈值。即前方红外测距传感器 GetIRDistance(4) 和 前方火焰传感器 GetAD(5)接近火焰时候的阈值。将灭火机器人正对蜡烛放置到足够近以保证机器人可以开启风扇灭火。如图 21；正常运行中此时机器人即可停止行驶开启风扇灭火。

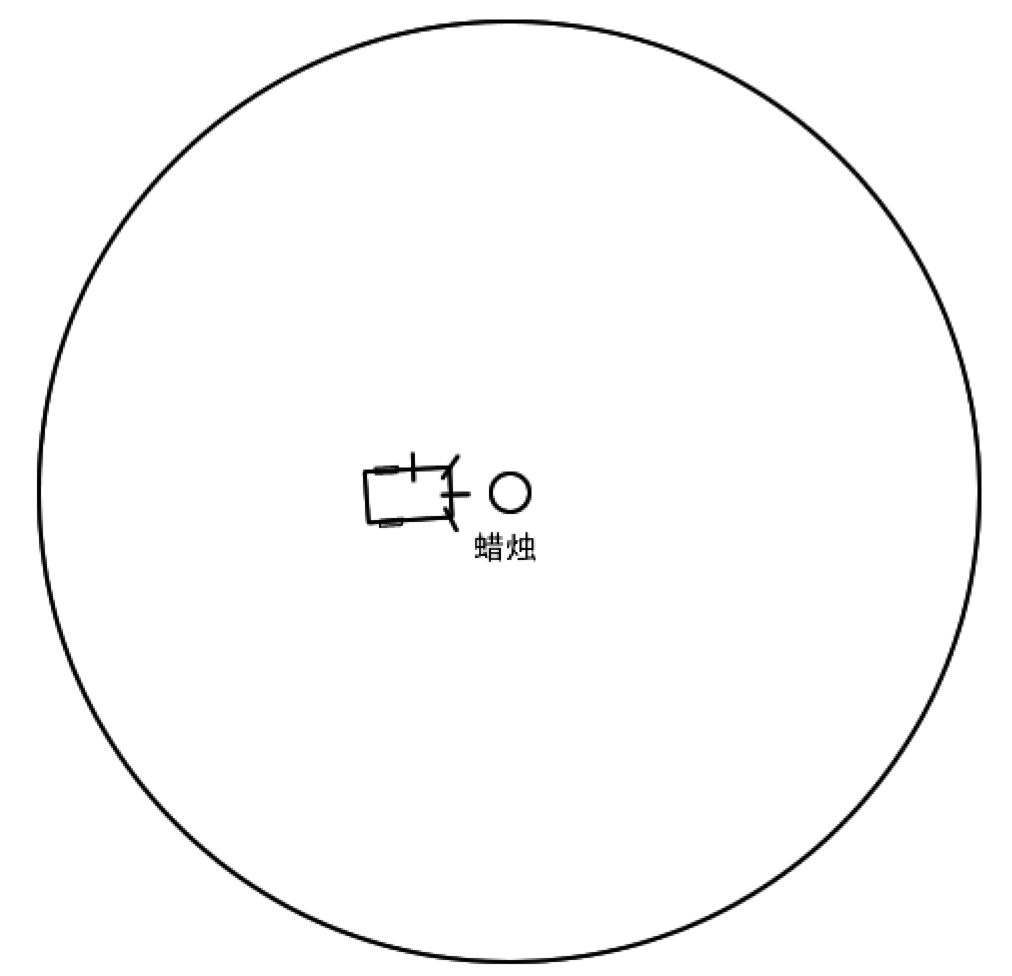


图 21 调试前方火焰传感器和红外测距传感器阈值

至此，机器人程序的几个阈值即可获得。

不同机器人的传感器会有个体差异，具体参数需要调试测试，从而保证机器人能自动运行。

在实际应用中，需要安装灭火风扇和高速灭火电机用来灭火。另外在机器人的右侧再安装一个火焰传感器亦可提高机器人发现火焰的能力。

至此。整个灭火机器人的制作完毕。

以上篇幅所用 MJ3100 控制器和 RobotDC 的具体用法请查阅对应的说明书。